PTO/SB/17 (07-07)
Approved for use through 06/30/2010. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Effective on 12/08/2004.
Fees pursuant to the Consolidated Appropriations Act, 2005 (H.R. 4818).

FEE TRANSMITTAL
For FY 2007

☐ Applicant claims small entity status. See 37 CFR 1.27

TOTAL AMOUNT OF PAYMENT	($)	500

Complete if Known	
Application Number	10/716,286
Filing Date	11/18/2003
First Named Inventor	Sriram Devanathan
Examiner Name	Farhan M. Syed
Art Unit	2165
Attorney Docket No.	592-L

METHOD OF PAYMENT (check all that apply)

☐ Check ☐ Credit Card ☐ Money Order ☐ None ☐ Other (please identify): _____

☑ Deposit Account Deposit Account Number: 19-3790 Deposit Account Name: Unisys Corporation

For the above-identified deposit account, the Director is hereby authorized to: (check all that apply)

☐ Charge fee(s) indicated below ☐ Charge fee(s) indicated below, **except for the filing fee**

☑ Charge any additional fee(s) or underpayments of fee(s) under 37 CFR 1.16 and 1.17 ☑ Credit any overpayments

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

FEE CALCULATION

1. BASIC FILING, SEARCH, AND EXAMINATION FEES

Application Type	FILING FEES Fee ($)	FILING FEES Small Entity Fee ($)	SEARCH FEES Fee ($)	SEARCH FEES Small Entity Fee ($)	EXAMINATION FEES Fee ($)	EXAMINATION FEES Small Entity Fee ($)	Fees Paid ($)
Utility	300	150	500	250	200	100	
Design	200	100	100	50	130	65	
Plant	200	100	300	150	160	80	
Reissue	300	150	500	250	600	300	
Provisional	200	100	0	0	0	0	

2. EXCESS CLAIM FEES

Fee Description	Fee ($)	Small Entity Fee ($)
Each claim over 20 (including Reissues)	50	25
Each independent claim over 3 (including Reissues)	200	100
Multiple dependent claims	360	180

Total Claims	Extra Claims	Fee ($)	Fee Paid ($)
_____ - 20 or HP =	_____ x	_____ =	_____

HP = highest number of total claims paid for, if greater than 20.

Indep. Claims	Extra Claims	Fee ($)	Fee Paid ($)
_____ - 3 or HP =	_____ x	_____ =	_____

HP = highest number of independent claims paid for, if greater than 3.

Multiple Dependent Claims Fee ($)	Multiple Dependent Claims Fee Paid ($)
_____	_____

3. APPLICATION SIZE FEE

If the specification and drawings exceed 100 sheets of paper (excluding electronically filed sequence or computer listings under 37 CFR 1.52(e)), the application size fee due is $250 ($125 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).

Total Sheets	Extra Sheets	Number of each additional 50 or fraction thereof	Fee ($)	Fee Paid ($)
_____ - 100 =	_____ / 50 =	_____ (round **up** to a whole number) x	_____ =	_____

4. OTHER FEE(S)

	Fees Paid ($)
Non-English Specification, $130 fee (no small entity discount)	
Other (e.g., late filing surcharge): Appeal Brief	500

SUBMITTED BY

Signature	Phuong-Quan Hoang	Registration No. (Attorney/Agent) 41,839	Telephone 949-380-5643
Name (Print/Type)	Phuong-Quan Hoang		Date 08/13/2007

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application. No.	:	**10/716,286**	Confirmation No. 6071
Inventor(s)	:	Sriram Devanathan *et al*	
Filed	:	November 18, 2003	
TC / Art Unit	:	2165	
Examiner	:	Syed, Farhan M.	
Docket No.	:	592-L	
Customer No.	:	34225	

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF

Dear Sir:

Applicant submits, the following Appeal Brief pursuant to 37 C.F.R. § 41.37 for consideration by the Board of Patent Appeals and Interferences. Please charge any additional fees or credit any overpayment to our deposit Account No.19-3790. A duplicate copy of the Fee Transmittal is enclosed for this purpose.

08/21/2007 HVUONG1 00000099 193790 10716286
02 FC:1402 500.00 DA

TABLE OF CONTENTS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE 1

I.REAL PARTY IN INTEREST 3

II.RELATED APPEALS AND INTERFERENCES 3

III.STATUS OF CLAIMS 3

IV.STATUS OF AMENDMENTS 3

V.SUMMARY OF CLAIMED SUBJECT MATTER 3

VI.GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL 10

VII.ARGUMENTs 10

A. Claims 15-26 Are Not Unpatentable under 35 U.S.C. §101 10

B. Claims 27-38 Are Not Unpatentable under 35 U.S.C. §101 11

C. Claims 1, 13, 15, 27 Are Not Anticipated by Teorey. 13

D. Claims 1, 13, 15, 27 Are Not Anticipated by Farpinyo. 15

E. Claims 1, 13, 15, 27 Are Not Anticipated by Shinjo. 19

F. Claims 2-12, 14, 16-26, 28-38 Are Not Unpatentable Over Teorey in view of Farpinyo. 22

VIII. CONCLUSION 25

IX.CLAIM APPENDIX 26

XI.EVIDENCE APPENDIX 39

XII.RELATED PROCEEDINGS APPENDIX 39

I. REAL PARTY IN INTEREST

The real party in interest is the assignee, Unisys Corporation.

II. RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences known to the appellants, the appellants' legal representative, or assignee, which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

III. STATUS OF CLAIMS

Claims 1-38 of the present application are pending. Claims 1-38 remain rejected. The Applicants hereby appeals the rejection of claims 1-38.

IV. STATUS OF AMENDMENTS

On October 12, 2006, Applicant filed an amendment, in response to a first Office Action issued on May 12, 2006. On January 11, 2007, the Examiner issued a Final Office Action. On April 11, 2007, Applicant filed a Notice of Appeal and a Pre-Appeal Brief Request For Review in response to the Final Office Action. No amendments to the claims have been filed subsequent to the Final Office Action. On May 3, 2007, the Pre-Appeal Review Panel issued a Notice of Panel Decision from Pre-Appeal Brief Review stating that the application remains under appeal with claims 1-38 rejected.

V. SUMMARY OF CLAIMED SUBJECT MATTER

1. Independent claims 1, 13, 15, and 27:

Database designs typically have two levels of information, which are logical information and physical information. In the Common Warehouse Model (CWM), the logical information or aspects are represented by entity-relationship (ER) diagrams, while the physical aspects are represented by relational elements in the relational design process where the end product is the structure of the database itself. Many of the terms from the ER world (logical) have near-equivalents in the relational world (physical). The following

ER terms (logical) in descending hierarchical order: model library, model, entity, attribute, are near-equivalents of these relational terms (physical): catalog, schema, table, column. The data to be transformed is stored in the common warehouse model (CWM) which defines a structure for the data[1].

Logical information, also called logical aspects, of the CWM are transformed into database design elements that are outputted to a database design tool[2].

Physical information, also called physical aspects, of the CWM are transformed into elements that are outputted to a database management system[3].

Since related items with the same names can be found in both the CWM (i.e., the input) and a database design (i.e., the output), a prefix will be added to the names of related items to distinguish them from one another. Only the ER (i.e., logical) and the relational (i.e., physical) models will be used as inputs. Items in the CWM are referred to as ER <name> for items in logical aspects, or relational <name> for items in physical aspects. Some items from the CWM that are common to both ER and relational worlds are prefixed by CWM. For the output, logical elements typically found in design tools with logical modeling support are referred to as design <name>. Physical elements typically found in a DBMS or in the physical modeling of DBMS provided in database design tools are referred to as DBMS <name>[4].

The CWM conversion system 45 reads the CWM representations 203 stored in the storage system 202 via the Application Programming Interface (API) 204 of the storage system 202, and transform the CWM representations into corresponding relational database elements. The type of data to be read from the storage system 202 and the format of the output of the CWM conversion system depend on the type of the output recipient. If the output recipient is a database design tool 206, the CWM conversion system 45 transforms the logical aspects of the CWM into design items in a design tool for a relational database. The CWM conversion system 45 communicates with the database design tool 206 via the API 208 of the tool 206. If the output recipient is a database management system (DBMS) 210, the CWM conversion system 45 transforms the physical aspects of the CWM to corresponding database management system (DBMS) representation in a relational database. The CWM conversion system 45 communicates

[1] See Specification, page 6, lines 17-28.
[2] See Specification, page 6, lines 7-9; page 7, lines 2-6.
[3] See Specification, page 6, lines 10-12; page 7, lines 7-12.

with the database management system (DBMS) 210 via the API 212 of the DBMS 210. In one embodiment, the CWM conversion system 45 can also read both the CWM logical aspects and physical aspects and produces the two respective outputs[5].

The CWM conversion system 45 processes a CWM representation in a hierarchical manner in order to transform the CWM representation into elements of a relational database. The CWM conversion system 45 processes the topmost (logical or physical) elements first. The high-level structure generated so far is outputted to the database design tool 206 or the DBMS 210. This sets up the framework for the output of the rest of the conversion of the CWM[6].

The logical aspects of the CWM comprise entity-relationship (ER) libraries. Each of the ER libraries comprises ER models. The corresponding design items in the relational database comprises design libraries, each of the design libraries comprises design models[7].

The physical aspects of the CWM comprise relational catalogs. Each of the relational catalogs comprises relational schemas. The corresponding DBMS representation comprises DBMS catalogs. Each of the DBMS catalogs comprises DBMS schemas[8].

2. Dependent claims 2-12, 14, 16-26 and 28-38:

Process 300 converts the logical aspects of a CWM to corresponding design items in a relational database. Process 300 scanned the ER libraries. For each of the ER libraries, that is being scanned, process 300 creates a corresponding design library in the relational database. For each of the ER models in the ER library being scanned, process 300 creates a corresponding design model in the corresponding design library to hold the corresponding information. Process 300 processes each of the ER models to produce the corresponding information for the corresponding design model. Process 300 processes each of the ER models independently of the other ER models. After all the ER models are processed, process 300 determines if there are any references between the ER models, that is, references across the boundaries of the ER models. If there are any such cross-model

[4] See Specification, page 7, lines 13-25.
[5] See Specification, page 12, lines 4-21; Figure 2.
[6] See Specification, page 12, lines 22-28; Figure 2.
[7] See Specification, page 12, line 30 - page 13, line 3; Figure 3.
[8] See Specification, page 19, lines 19-22; Figure 12.

references, process 300 makes corresponding cross-model references in the corresponding design models[9].

Process 400 processes an ER model to produce a corresponding design model for the relational database. Process 400 processes the ER subject areas included in the ER model. Subject areas constitute a way of organizing the tables for understanding purposes (note that tables are only linked to the subject areas, not included therein). For this reason, subject areas can only exist in the logical aspects of the CWM, not in the physical aspects of the CWM. It is possible for a CWM to have no subject area. Process 400 processes the ER domains included in the ER model. Process 400 processes domain inheritance for each of the ER domains, then processes ER entities included in the ER model. Process 400 processes the entity subtype relationships for each of the ER entities, and processes the non-subtype relationships for each of the ER relationships[10].

Process 500 processes each of the subject areas included in an ER model. Process 500 goes through the list of the ER subject areas and creates, for each ER subject area included in the ER model, a corresponding design subject area in the corresponding design model to represent that ER subject area. The design subject area includes all the properties of the ER subject area[11].

Process 600 processes each of the ER domains included in an ER model. For each ER domain that exists, process 600 creates a corresponding design domain to represent this ER domain. Process 600 obtains the parameters for the ER domain, including the base type, default and constraint. Process 600 uses this information to set the corresponding parameters for the design domain. It is noted that, domains in general and particularly those related to textual base types are sometimes placed in a separate ER model. Other ER models would reference this separate ER model if needed information are to be found there. This is an example of the cross-model references that need to be resolved after all the ER libraries have been processed[12].

Process 700 processes the ER domain inheritance of each of the ER domains included in an ER model. Process 700 sets a first pointer to the first ER domain in the ER

[9] See Specification, page 13, lines 5-17; Figure 3.
[10] See Specification, page 14, lines 3-18; Figure 4.
[11] See Specification, page 14, lines 19-23, lines 28-29; Figure 5.
[12] See Specification, page 15, lines 1-16; Figure 6.

model. Process 700 checks whether the ER domain exists. If it does not exist, process 700 terminates. Otherwise, process 700 sets a second pointer to the first CWM generalization that links this ER domain. If there is no such generalization, process 700 increases the first pointer to point to the next ER domain. If there is such CWM generalization, process 700 determines which are the parent and child ER domains for this CWM generalization, that is, the ends of the link of which the ER domain in question represents one end. The parent and child ER domains correspond to corresponding parent and child design domains in the relational database. To represent the generalization in the relational database, process 700 creates an inheritance link from the corresponding child design domain to the corresponding parent design domain. This inheritance link in the relational database corresponds to the inheritance link in the CWM[13].

Process 800 processes entities included in an ER model. Process 800 sets a pointer to the first ER entity in the ER model. If this ER entity exists, process 800 creates for this ER entity a corresponding design entity in the design model that corresponds to the ER model. Process 800 obtains the list of all the ER subject areas that include this ER entity as a member. The ER subject areas have corresponding design subject areas in the relational database. Process 800 adds the corresponding design entity as a member of the corresponding design subject areas. Process 800 invokes process 900 to process the attributes associated with the ER entity[14].

Process 900 processes the attributes associated with an ER entity. Process 900 sets a pointer to the first ER attribute specific to the ER entity. If this ER attribute exists, process 900 creates a design attribute to represent this ER attribute. Process 900 attaches the design attribute to the design entity that was created by process 800 to correspond to this ER entity. Process 900 sets the "type" reference of the design attribute based on the "type" reference of the ER attribute. Note that the "type" reference of the ER attribute links the ER attribute to an ER domain. This ER domain in the CWM has a corresponding design domain in the relational database. By setting the "type" reference of the design attribute, process 900 links the design attribute to this corresponding design domain. It is noted that the "type" reference can point to either one of the design domains previously created or a basic data type supported by the database design tool. Process 900 determines

[13] See Specification, page 15, line 18 through page 16, line 4; Figure 7.
[14] See Specification, page 16, lines 7-18; Figure 8.

whether the ER attribute is part of the ER primary key associated with the ER entity. If the ER attribute is part of the ER primary key, process 900 flags the design attribute as part of the design primary key associated with the design entity. Process 900 then proceeds to process the next ER attribute[15].

Process 1000 processes the entity subtype relationships for an ER entity included in an ER model. Process 1000 sets a pointer to the first CWM generalization that links the ER entities of this ER model. Process 1000 determines whether the CWM generalization exists. If it exists, process 1000 determines which are the parent and child ER entities for this generalization. The parent and child ER entities correspond to corresponding parent and child design entities in the relational database. To represent this generalization in the relational database, process 1000 creates an inheritance link from the corresponding child design entity to the corresponding parent design entity[16].

Process 1100 processes the entity non-subtype relationships in an ER model. Process 1100 sets a pointer to the first ER relationship in the ER model. Process 1100 checks whether the ER relationship exists. If it does not exist, process 1100 terminates. Otherwise, process 1100 obtains the references to the parent and child ER entities for this ER entity non-subtype relationship. The parent and child ER entities are represented by corresponding parent and child design entities in the relational database. To represent the ER non-subtype relationship in the relational database, process 1100 creates links between the corresponding child design entity (or entities) and the corresponding parent design entity or entities. Due to the wide diversity in database design tools, this link could be established directly as a two-way link or by the creation of a design relationship to which each of the involved design entities is linked. Process 1100 sets the cardinality of the design relationship and sets the relationship type to "identifying" or "non-identifying" based on information stored as CWM "tagged elements" associated with the ER relationship. The cardinality could be "1 to 1" or "n to 1" where n is an integer (can be zero). Process 1100 examines the ends of this ER relationship to determine whether this ER relationship has any referential rule. Referential rules are also called integrity constraints.

[15] See Specification, page 16, line 22 through page 17, line14; Figure 9.
[16] See Specification, page 17, lines 15-1627; Figure 10.

If there is any referential rule, process 1100 proceeds to process the associated referential rule or rules[17].

Process 1100 obtains the values of the referential rules including "Insert", "Update", and "Delete" from the CWM, then sets the values of the corresponding referential rules for the design link (or design relationship) that corresponds to the ER relationship. Process 1100 scans the child ER entity to determine whether any of the ER attributes of the child ER entity has migrated from the parent ER entity. If there are ER attributes in the child ER entity that have migrated from the parent ER entity, then process 1100 creates a design foreign key under the child design entity. It is noted that only one foreign key is created for all the attributes that have migrated from the parent entity to the child entity. Process 1100 creates references to the design attributes that correspond respectively to the ER attributes that have migrated. It is noted that all of the ER items identified above may have their own diagram and annotated text information attached to them. In such case, these diagrams and annotated text information are also stored for the corresponding design items[18].

Process 1200 converts the physical aspects of a common warehouse model (CWM) to corresponding database management system (DBMS) representation in a relational database. The physical aspects of the CWM comprise relational catalogs. Each of the relational catalogs comprises relational schemas. The corresponding DBMS representation comprises DBMS catalogs. Each of the DBMS catalogs comprises DBMS schemas. Process 1200 can output the DBMS representation to a DBMS. This output is typically in the form of a script containing instructions to modify existing or create new elements in the relational database. For example, the script used in Structured Query Language (SQL) could be the form of this output. Process 1200 scans through the relational catalogs. For each of the relational catalogs, that is being scanned, process 1200 creates a corresponding DBMS catalog to be outputted to the DBMS. For each of the relational schemas in each of the relational catalogs, process 1200 creates a corresponding DBMS schema in the corresponding DBMS catalog to hold the corresponding information. Process 1200 processes each of the relational schemas to produce the corresponding information for the

[17] See Specification, page 17, line 28 through page 18, line20; Figure 11A.
[18] See Specification, page 18, line 21 through page 19, line 5; Figure 11B.

corresponding DBMS schema. Process 1200 processes each of the relational schemas in the CWM independently of the other relational schemas[19].

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Claims 15-26 and 27-38 stand rejected under 35 U.S.C. §101 as being unpatentable for being directed to non-statutory subject matter.

Claims 1, 13, 15, 27 stand rejected under 35 U.S.C. §102(b), as being anticipated by a non-patent literature titled "A Logical Design Methodology for Relational Database Using the Extended Entity-Relationship Model" by Toby J. Teorey *et al*, ACM Computing Survey (CSUR), June 1986, vol. 18, issue 2 (hereinafter, "Teorey").

Claims 1, 13, 15, 27 further stand rejected under 35 U.S.C. §102(b), as being anticipated by a non-patent literature titled "Designing and Creating Relational Schemas with a CWM-Based Tool" by Kumpon Farpinyo *et al* (hereinafter, "Farpinyo").

Claims 1, 13, 15, 27 further stand rejected under 35 U.S.C. §102(e), as being anticipated by U.S. Patent Application Publication 2004/0133581 A1 of Shinjo (hereinafter, "Shinjo").

Claims 2-12, 14, 16-26, 28-38 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Teorey in view of Farpinyo.

VII. ARGUMENTS

A. Claims 15-26 Are Not Unpatentable under 35 U.S.C. §101.

In the Final Office Action, the Examiner maintained his rejection of claims 21, 23-40 under 35 U.S.C. §101, stating "The Examiner disagrees with the Applicant's analysis of a carrier wave. The Examiner is not refuting the use of RF waves, but instead the use of the words carrier wave or a signal modulated by a carrier. The Examiner will continue to rely on the Interim Guidelines, as explained in the previous office action dated 12 May

[19] See Specification, page 19, line 16 through page 20, line 5; Figure 12.

2006, as the basis for rejection of claims 15 and 21" (Final Office Action, pages 2-3, item 5).

In response to the first Office action, Applicant has amended claim 15 to limit claims 15 and its dependent claims to machine-accessible storage medium in order to obtain a timely Notice of Allowance.

The Examiner repeated the rejection without taking note of the Applicant's amendment as presented in the previously filed response. The MPEP requires that the Examiner's action will be complete as to all matters. 37 CRF 1.104; MPEP 707.07. Since the Examiner's action in the Office Action is incomplete in that there is no answer to the substance of Applicant's amendment previously presented, the rejections have been improperly made.

B. Claims 27-38 Are Not Unpatentable under 35 U.S.C. §101.

In the Final Office Action, the Examiner maintained his rejection of system claims 27-38 under 35 U.S.C. §101 without responding to Applicant's arguments presented in the response filed on October 12, 2006, page 25, item 2 (lines 7-22). In addition, the Examiner erroneously referred to independent claim 27 as claim 21 and dependent claims 28-38 as claims 25-32 in both the first Office Action of May 12, 2006 (pages 10-11) and the Final Office Action (page 3, line 4).

Specifically, the Examiner states that Claim 27 recites "a memory coupled to the processor, the memory containing program code that, when executed by the processor, causes the processor to perform the operation". The Examiner states that the word "program code" is used on page 10 of the Specification which recites that the program code can be stored in a process, or a machine accessible, or transmitted by a computer data signal embodied in a carrier wave, etc. The Examiner then states that "implementing the claim would render the result of the claim as intangible", that "[a] signal-bearing medium is not tangible, and cannot tangibly embody a computer program or process since a computer cannot understand/realize (i.e., execute) the computer program or process when embodied on the data signal", and concludes that "a data signal does not meet the "useful, concrete, and tangible" requirement (First Office Action, page 10 line 8 through page 11, line 8).

Applicant respectfully submits that the Examiner's rejection is clearly erroneous. Claim 27 is a system claim that clearly recites a processor, a memory coupled to the processor, the memory containing program code. It is clear from the claim that the program code is contained in the memory. Applicant fails to understand how "implementing the claim would render the result of the claim as intangible" and how the Examiner could find in Claim 27 a transmission medium or a data signal. Accordingly, Applicant respectfully submits that claims 27-38 are patentable under 35 U.S.C. §101.

In response to Applicant's arguments above regarding the 35 U.S.C. §101 rejection of system claims 27-38, the Examiner stated the following: "The Examiner disagrees with the Applicant's analysis of a carrier wave. The Examiner is not refuting the use of RF waves, but instead the use of the words carrier wave or a signal modulated by a carrier. The Examiner will continue to rely on the Interim Guidelines, as explained in the previous office action, dated 12 May 2006, as the basis for rejection of claims 15 and 21" (Final Office Action, pages 2-3, item 5).

The Examiner did not respond to Applicant's arguments regarding the system claims 27-38. Accordingly, the Examiner's lack of response to these arguments amounts to an improper office action.

Where a claim is refused for any reason relating to the merits thereof it should be "rejected" and the ground of rejection fully and clearly stated. See MPEP 707.07(d). Where the applicant traverses an objection, the Examiner should, if he or she repeats the rejection, take note of the applicant's argument and answer the substance of it. See MPEP 707.07(f). It is important for an examiner to properly communicate the basis for a rejection so that the issues can be identified early and the applicant can be given fair opportunity to reply. See MPEP 706.02(j). The Examiner repeated the rejection without taking note of the Applicant's argument and without answering the substance of Applicant's argument as presented in the previously filed response. The MPEP requires that the Examiner's action will be complete as to all matters. 37 CRF 1.104; MPEP 707.07. Since the Examiner's action in the Office Action is incomplete in that there is no answer to the substance of Applicant's arguments previously presented, the rejections have been improperly made.

C. Claims 1, 13, 15, 27 Are Not Anticipated by Teorey.

In the Final Office Action, the Examiner maintained the rejection of claims 1, 13, 15, 27 under 35 U.S.C. §102(b), as being anticipated by a non-patent literature titled "A Logical Design Methodology for Relational Database Using the Extended Entity-Relationship Model" by Toby J. Teorey *et al* ("Teorey"). Applicant respectfully traverses the rejection and submits that the Examiner has not met the burden of establishing a *prima facie* case of anticipation.

Teorey discloses a methodology for the design of large relational databases. First, the data requirements are conceptualized using an extended entity-relationship model, with the extensions being additional semantics such as ternary relationships, optional relationships, and the generalization abstraction. The extended entity-relationship model is then decomposed according to a set of basic entity-relationship constructs, and these are transformed into candidate relations (Teorey, Abstract page 1; Conclusion, page 220).

The Examiner rejected claims 1, 13, 15, 27 by simply citing the Abstract of Teorey without specifically identifying any element in Teorey that the Examiner considered as anticipating an element of the claims.

Teorey does not disclose, either inherently or explicitly, at least one of the following elements: converting logical aspects of a common warehouse model (CWM) to corresponding design items for a relational database by processing in a hierarchical manner the logical aspects and creating the corresponding design items, the logical aspects comprising entity-relationship (ER) libraries, the ER libraries comprising ER models, the corresponding design items comprising design libraries, the design libraries comprising design models.

It is not possible for Teorey to refer to CWM since CWM did not exist at the time Teorey was published, namely 1986, more than 10 years before CWM was introduced.

To anticipate a claim, the reference must teach every element of the claim. "A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." Vergegaal Bros. v. Union Oil Co. of California, 814 F.2d 628, 631, 2 USPQ 2d 1051, 1053 (Fed. Cir. 1987). "The identical invention must be shown in as complete detail as is contained in the...claim." Richardson v. Suzuki Motor Co., 868 F.2d 1226, 1236, 9 USPQ 2d 1913,

1920 (Fed. Cir.1989). Since the Examiner failed to show that Teorey teaches or discloses any of the elements of the claims, the rejection under 35 U.S.C. §102 is improper.

In response to Applicant's arguments, the Examiner stated that CWM is an instance of a data warehouse model and gave the definition of a data warehouse model as defined in the Microsoft Computer Dictionary, 5th Edition. The Examiner then stated that "Furthermore, the Applicant states in the specification that CWM stems from UML, which is an instance of object-oriented programming. Object-oriented programming has existed from the 1970s. Given that the Examiner is allowed the broadest interpretation of the claims, the prior art of record clearly anticipates the recited claims. Therefore, the rejection with this cited prior art is sustained." (Final Office Action, page 4, first full paragraph). The Examiner then repeated exactly what the Examiner had written in the First Office Action, that is, rejecting claims 1, 13, 15, 27 by simply citing the Abstract of Teorey without specifically identifying any element in Teorey that the Examiner considered as anticipating an element of the claims.

Applicant respectfully disagrees for the following reasons.

First, the Examiner did not follow the requirements prescribed by case law for a rejection under 35 U.S.C. §102. The Examiner rejected claims 1, 13, 15, 27 by simply citing the Abstract of Teorey without specifically identifying any element in Teorey that the Examiner considered as anticipating an element of the claims.

Second, the Examiner did not respond to Applicant's argument that case law requires that "to anticipate a claim, the reference must teach every element of the claim".

Third, the Examiner's response that Teorey, which predates CWM by more than 10 years, can anticipate the claims since CWM stems from UML, and UML is an instance of object-oriented programming, and object-oriented programming has existed since the 1970s, is clearly erroneous. This clearly erroneous response amounts to stating that any data warehouse model that is related to and came after object-oriented programming can be found to be anticipated by Teorey. Following the logic of the Examiner's erroneous response would lead to the clearly erroneous conclusion that CWM itself can also be found anticipated by Teorey.

D. Claims 1, 13, 15, 27 Are Not Anticipated by Farpinyo.

In the Final Office Action, the Examiner rejected claims 1, 13, 15, 27 under 35 U.S.C. §102(b), as being anticipated by a non-patent literature titled "Designing and Creating Relational Schemas with a CWM-Based Tool" by Kumpon Farpinyo *et al*, Department of Computer Engineering, Chulalongkorn University, Bangkok, Thailand (hereinafter, "Farpinyo"). Applicant respectfully traverses the rejection and submits that the Examiner has not met the burden of establishing a prima facie case of anticipation.

Farpinyo discloses that a tool called ER2CWM can create CWM relational database schemas from physical data models represented by ER diagrams (Abstract). The tool supports the creation of ER diagrams through its ER editor, generates CWM Relational metadata, and creates database schemas. It can also transform database schemas back into CWM Relational metadata and ER diagrams respectively (Farpinyo, page 457, last 3 lines, page 458, first 2 lines).

Farpinyo discloses that the tool *ER2CWM considers only the part of CWM for relational database schemas called CWM Relational* (Farpinyo, Section 1, page 456, last 2 lines) (emphasis added). As well known in the art, CWM Relational is the part that contains the physical information. Refer, for example, to this OMG weblink:

http://www.omg.org/docs/formal/03-03-29.pdf

Farpinyo gave a brief introduction to CWM Relational by explaining several elements of CWM Relational, such as Catalog, Schema, Table, Column, InitialValue, CheckConstraint, PrimaryKey, ForeignKey, SQL data type (Farpinyo, Section 2, page 458).

In Section 3 (Farpinyo, page 459), Farpinyo discloses that the tool ER2CWM is composed of four modules: ER Editor, Metadata, DBMS Information, and ER Module. The ER Editor is the editor for designing physical data models with ER diagrams. The ER Editor is a GUI for user to create CWM Relational metadata, select DBMSs to create database schemas, or create CWM Relational metadata and ER diagrams from existing relational databases. The Metadata module creates and maintains two types of metadata: diagram metadata and CWM metadata. Diagram metadata (DIA) is the metadata of ER models with display information for the diagrams. CWM metadata represents ER models or database schemas and conforms to CWM v1.0. The DBMS Information module creates database schemas from CWM Relational metadata, reads in existing database schemas to

create CWM Relational metadata and ER diagrams, and maintains information about DBMSs that the tool supports. The ER Module connects together the other three modules. It contacts DBMS Information module when database designers select DBMSs to design physical data models or to create database schemas. It interacts with the Metadata module to get and save DIA and CWM Relational metadata. (Farpinyo, Section 3, page 459).

Note that, although Farpinyo uses the term "CWM metadata" in the description of the Metadata module, Farpinyo states that the ER Module interacts with the Metadata module to get and save DIA and CWM Relational metadata. Thus, what is called CWM metadata in the description of the Metadata module is actually CWM Relational metadata. Furthermore, Farpinyo clarifies that the tool "ER2CWM cannot properly display ER diagrams that are not created by the tool itself. That is, ER diagrams that are created from any CWM documents or created from pre-existing database schemas do not have associated .DIA files that ER2CWM needs for a proper display. Database designers will have to arrange the layout of the diagrams by themselves." (Farpinyo, page 460, footnote 1). Since the tool cannot display ER diagrams created from any CWM documents, this shows that Farpinyo does not use the logical information part of CWM.

In Section 4 (Farpinyo, page 460), an example is given with an ER diagram, CWM metadata, and generated database schema as results of using the tool. A database designer first selects Sybase Adaptive Server (a DBMS) as a target of the design, and draws an ER diagram on the GUI of the tool. The tool then generates a corresponding CWM Relational metadata for this design. The designer later change to create a database schema for MS SQL Server (another DBMS) instead by using the CWM metadata generated earlier.

Farpinyo discloses that a user can specify a target DBMS and can input an ER diagram to the ER2CWM tool by drawing it using the GUI of the tool. The ER2CWM tool then uses the ER diagram to generate and store a corresponding CWM Relational metadata. When the user later specifies a different DBMS, the ER2CWM tool uses the stored CWM Relational metadata to generate a database schema for this different DBMS (Farpinyo, page 460, Section 4).

Farpinyo discloses a user manual of the ER2CWM tool to show how to use the GUI of the ER2CWM tool to create an ER diagram (Farpinyo, User Manual, page 2 through page 7, first paragraph); to create a database schema from the ER diagram Farpinyo, User Manual, page 7); to read a schema from a specified database (Farpinyo,

User Manual, pages 8-9); to create an ER diagram from a selected CWM Metadata file (Farpinyo, User Manual, pages 9-10); to save an ER diagram into HTML format (Farpinyo, User Manual, page 11, item 5); and to print an ER diagram (Farpinyo, User Manual, page 11, item 6).

Farpinyo only deals with the CWM Relational, that is, the physical aspects of CWM, not the logical aspects of CWM. Therefore, Farpinyo cannot anticipate claims 1, 13, 15, 27 since claims 1, 15, 27 are directed to converting logical aspects of CWM to design elements, and claim 15 is directed to both converting logical aspects of CWM to design elements and converting physical aspects of CWM to DBMS items.

The Examiner rejected claims 1, 13, 15, 27 by simply citing the Abstract and the first paragraph of the Introduction section of Farpinyo without specifically identifying any element in Farpinyo that the Examiner considered as anticipating an element of the claims.

Farpinyo does not disclose, either inherently or explicitly, any of the following elements: converting logical aspects of a common warehouse model (CWM) to corresponding design items for a relational database by processing in a hierarchical manner the logical aspects and creating the corresponding design items, the logical aspects comprising entity-relationship (ER) libraries, the ER libraries comprising ER models, the corresponding design items comprising design libraries, the design libraries comprising design models.

To anticipate a claim, the reference must teach every element of the claim. "A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." Vergegaal Bros. v. Union Oil Co. of California, 814 F.2d 628, 631, 2 USPQ 2d 1051, 1053 (Fed. Cir. 1987). "The identical invention must be shown in as complete detail as is contained in the...claim." Richardson v. Suzuki Motor Co., 868 F.2d 1226, 1236, 9 USPQ 2d 1913, 1920 (Fed. Cir.1989). Since the Examiner failed to show that Farpinyo teaches or discloses any of the elements of the claims, the rejection under 35 U.S.C. §102 is improper.

In response to Applicant's arguments, the Examiner stated that "the methodology of the use of ER2CWM is clearly anticipated by prior art of record, wherein Sections 3 and 4 clearly illustrate the use of ER2CWM. An ordinary person skilled in the art clearly understands that the steps required to execute a tool like ER2CWM to create CWM

relational database schemas from physical data models represented by ER diagrams must include the steps that are recited in claims 1, 21, 41. This methodology is fundamental, let alone the essence of creating relational schemas with a CWM-based tool." (Final Office Action, page 4-page 5, item 2). Applicant would like to point out that the Examiner has referred to the wrong claims. The claims at issue are 1, 13, 15, 27, not 21 and 41.

Applicant respectfully disagrees for the following reasons.

First, the Examiner did not follow the requirements prescribed by case law for a rejection under 35 U.S.C. §102. The Examiner rejected claims 1, 13, 15, 27 by simply citing the Abstract and the first paragraph of the Introduction section of Farpinyo without specifically identifying any element in Farpinyo that the Examiner considered as anticipating an element of the claims.

Second, the Examiner did not respond to Applicant's argument that case law requires that "to anticipate a claim, the reference must teach every element of the claim".

Third, the Examiner stated that "the methodology of the use of ER2CWM is clearly anticipated by prior art of record", but did not cite what prior art of record the Examiner had in mind.

Fourth, the Examiner did not respond to Applicant's argument that Farpinyo cannot anticipate claims 1, 13, 15, 27 since Farpinyo only deals with the CWM Relational, that is, the physical aspects of CWM, not the logical aspects of CWM. Accordingly, the Examiner's lack of response to this argument amounts to an improper office action.

Where a claim is refused for any reason relating to the merits thereof it should be "rejected" and the ground of rejection fully and clearly stated. See MPEP 707.07(d). Where the applicant traverses an objection, the Examiner should, if he or she repeats the rejection, take note of the applicant's argument and answer the substance of it. See MPEP 707.07(f). It is important for an examiner to properly communicate the basis for a rejection so that the issues can be identified early and the applicant can be given fair opportunity to reply. See MPEP 706.02(j). The Examiner repeated the rejection without taking note of the Applicant's argument and without answering the substance of Applicant's argument as presented in the previously filed response. The MPEP requires that the Examiner's action will be complete as to all matters. 37 CRF 1.104; MPEP 707.07. Since the Examiner's action in the Office Action is incomplete in that there is no

answer to the substance of Applicant's arguments previously presented, the rejections have been improperly made.

Therefore, Applicant submits that independent claims 1, 13, 15, 27 and their respective dependent claims are distinguishable over the cited prior art references.

E. Claims 1, 13, 15, 27 Are Not Anticipated by Shinjo.

In the Final Office Action, the Examiner rejected claims 1, 13, 15, 27 under 35 U.S.C. §102(e), as being anticipated by Shinjo (U.S. Patent Application Publication 2004/0133581 A1). Applicant respectfully traverses the rejection and submits that the Examiner has not met the burden of establishing a prima facie case of anticipation.

Shinjo discloses a database management system that includes an object conversion unit converting each of a plurality of natural objects and each of a plurality of object IDs of consecutive data according to a predetermined rule in a unique relationship and a bidirectional manner; and a database storing tables of a hierarchical structure including the object IDs converted by said object conversion unit as a permanent object holding data during a period (Shinjo, Abstract).

Citing paragraph [009]-[0010] of the Background Art section of Shinjo, the Examiner stated that "The preceding text clearly indicates that an E-R model is frequently used to convert data from a source into a relational database. It is well known in the art that when creating a logical or conceptual design that there exist models and libraries within the ER-Model which corresponds to the models and libraries of a relational database. Although the primary reference does not refer to CWM, it is an intended use to convert CWM information into a relational database through an ER Model" (Final Office Action, page 13, lines 1-6). Thus, the Examiner admits that Shinjo does not refer to CWM. Since Shinjo does not refer to CWM, Shinjo cannot teach any of the elements of claims 1, 13, 15, 27.

To anticipate a claim, the reference must teach every element of the claim. "A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." Vergegaal Bros. v. Union Oil Co. of California, 814 F.2d 628, 631, 2 USPQ 2d 1051, 1053 (Fed. Cir. 1987). "The identical invention must be shown in as complete detail as is contained in the...claim." Richardson v. Suzuki Motor Co., 868 F.2d 1226, 1236, 9 USPQ 2d 1913,

1920 (Fed. Cir.1989). Since the Examiner failed to show that Shinjo teaches or discloses any one of the above elements, the rejection under 35 U.S.C. §102 is improper.

In response to Applicant's argument that, since Shinjo does not refer to CWM, Shinjo cannot teach any of the elements of claims 1, 13, 15, 27, the Examiner stated the following: "When using any type of database system, whether it is hierarchical, network based, relational, or object-oriented, etc., the data stored in such databases may be manipulated through an object-oriented programming language, such as UML, XML, C++, Java, etc.via the use of SQL. Since the Examiner has already explained CWM as an instance of data warehouse modeling, which is clearly taught in this prior art of record (see at least paragraph [0009-0013], the Examiner clearly believes that CWM is implicitly anticipated. Thus, Shinjo teaches each and every element of claims 1, 13, 15, and 27 and therefore the rejections are sustained." (Final Office Action, page 5, item 3).

Applicant respectfully disagrees for the following reasons.

First, the Examiner did not follow the requirements prescribed by case law for a rejection under 35 U.S.C. §102. The Examiner rejected claims 1, 13, 15, 27 by merely citing paragraph [009]-[0010] of the Background Art section of Shinjo without specifically identifying any element in Shinjo that the Examiner considered as anticipating an element of the claims.

Second, the Examiner did not respond to Applicant's argument that case law requires that "to anticipate a claim, the reference must teach every element of the claim".

Third, it is not clear to Applicant what the Examiner meant by stating that "the Examiner clearly believes that CWM is implicitly anticipated" (Final Office Action, page 5, item 3). If the Examiner meant that CWM is implicitly anticipated by Shinjo, then this is clearly erroneous, since CWM was introduced before Shinjo and was recognized as a major invention.

Fourth, the Examiner merely states that "although the primary reference does not refer to CWM, it is an intended use to convert CWM information into a relational database through an ER Model" (Final Office Action, page 13, lines 4-6). Apparently, the Examiner applied the theory of inherency or relied on official notice to arrive at the conclusion that Shinjo anticipates claims 1, 13, 15, 27. Applicant submits that the Examiner's reliance of the theory of inherence or official notice is misplaced for the following reasons.

First, the fact that a certain result or characteristic may occur or be present in the prior art is not sufficient to establish the inherency of that result or characteristic. In re Rijckaert, 9 F.3d 1531, 1534, 28 USPQ2d 1955, 1957 (Fed. Cir. 193). "To establish inherency, the extrinsic evidence 'must make clear that the missing descriptive matter is necessarily present in the thing described in the reference, and that it would be so recognized by persons of ordinary skill. Inherency, however, may not be established by probabilities or possibilities. The mere fact that a certain thing may result from a given set of circumstances is not sufficient.'" In re Robertson, 169 F.3d 743, 745, 49 USPQ2d 1949, 1950-51 (Fed. Cir. 1999). "In relying upon the theory of inherency, the examiner must provide a basis in fact and/or technical reasoning to reasonably support the determination that the allegedly inherent characteristic necessarily flows from the teachings of the applied prior art." Ex parte Levy, 17 USPQ2d 1461, 1464 (Bd. Pat. App. & Inter. 1990) (emphasis in original). Here, Shinjo does not disclose any conversion of logical aspects of CWM to design elements for relational database. The Examiner has not shown that such conversion of logical aspects of CWM necessarily flows from the teachings of Shinjo.

Second, official notice unsupported by documenting evidence should only be taken by the Examiner where the facts asserted to be well-known, or to be common knowledge in the art are capable of instant and unquestionable demonstration as being well known. In re Ahlert, 424 F.2d 1088, 1091, 165 USPQ 418, 420 (CCPA 1970); MPEP 2144.03A. It would not be appropriate for the Examiner to take official notice of facts without citing a prior art reference. MPEP 2144.03A. Furthermore, if official notice is taken of a fact, unsupported by documentary evidence, the technical line of reasoning underlying a decision to take such notice must be clear and unmistakable. MPEP 2144.03B. Here, Shinjo neither discloses nor suggests a CWM conversion. The Examiner did not provide a technical line of reasoning which must be clear and unmistakable. The Examiner merely states that "although the primary reference does not refer to CWM, it is an intended use to convert CWM information into a relational database through an ER Model" (Final Office Action, page 13, lines 4-6). Therefore, the Examiner's reasoning is not clear and not unmistakable.

Therefore, Applicant submits that claims 1, 13, 15, 27 are distinguishable over the cited prior art reference.

F. Claims 2-12, 14, 16-26, 28-38 Are Not Unpatentable Over Teorey in view of Farpinyo.

In the Final Office Action, the Examiner rejected claims 2-12, 14, 16-26, 28-38 under 35 U.S.C. §103(e) as being unpatentable over a non-patent literature titled "A Logical Design Methodology for Relational Database Using the Extended Entity-Relationship Model" by Toby J. Teorey *et al*, ACM Computing Survey (CSUR), June 1986, vol. 18, issue 2 (hereinafter, "Teorey") in view of a non-patent literature titled "Designing and Creating Relational Schemas with a CWM-Based Tool" by Kumpon Farpinyo *et al*, Department of Computer Engineering, Chulalongkorn University, Bangkok, Thailand (hereinafter, "Farpinyo"). Applicant respectfully traverses the rejection and submits that the Examiner has not met the burden of establishing a *prima facie* case of obviousness.

The Supreme Court in *Graham v. John Deere*, 383 U.S. 1, 148 USPQ 459 (1966), stated: "Under § 103, the scope and content of the prior art are to be determined; differences between the prior art and the claims at issue are to be ascertained; and the level of ordinary skill in the pertinent art resolved. Against this background, the obviousness or nonobviousness of the subject matter is determined." MPEP 2141. In *KSR International Co. vs. Teleflex, Inc.*, 127 S.Ct. 1727 (2007) (Kennedy, J.), the Court explained that "[o]ften, it will be necessary for a court to look to interrelated teachings of multiple patents; the effects of demands known to the design community or present in the marketplace; and the background knowledge possessed by a person having ordinary skill in the art, all in order to determine whether there was an apparent reason to combine the known elements in the fashion claimed by the patent at issue." The Court further required that an explicit analysis for this reason must be made. "[R]ejections on obviousness grounds cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness." *KSR 127 S.Ct.* at 1741, quoting *In re Kahn*, 441 F.3d 977, 988 (Fed. Cir. 2006). In the instant case, Applicant respectfully submits that there are significant differences between the cited references and the claimed invention and there is no apparent reason to combine the known elements in the manner as claimed, and thus no *prima facie* case of obviousness has been established.

Teorey discloses a methodology for the design of large relational databases, as discussed above in Section C. Teorey does not refer to CWM.

Farpinyo discloses that a tool called ER2CWM can create CWM relational database schemas from physical data models represented by ER diagrams, as discussed in above in Section D. Farpinyo only uses the physical information part of CWM, not the logical information part. In other words, Farpinyo does not consider any of the ER elements recited in claims 2-12, 14, 16-26, 28-38.

Teorey and Farpinyo, taken alone or in any combination, do not disclose, suggest, or render obvious, at least one of the following elements: (1) converting logical aspects of a common warehouse model (CWM) to corresponding design items for a relational database by processing in a hierarchical manner the logical aspects and creating the corresponding design items, the logical aspects comprising entity-relationship (ER) libraries, the ER libraries comprising ER models, the corresponding design items comprising design libraries, the design libraries comprising design models; wherein converting comprises the operations of: (a) scanning through the ER libraries; (b) for a first of the ER libraries, creating a corresponding first design library; (c) for each of the ER models in the first ER library, creating a corresponding design model in the corresponding first design library to hold corresponding information; (d) processing each of the ER models to produce corresponding information for the corresponding design model; (e) determining if there are any references between the ER models; and (f) if there are any references between the ER models, specifying corresponding references in corresponding design models.

As discussed in Sections C and D above, neither Teorey nor Farpinyo discloses or suggests element (1) listed above. Accordingly, using a combination of Teorey and Farpinyo in rejecting claims 2-12, 14, 16-26, 28-38 which include element (1) is improper.

The Examiner failed to establish the factual inquires in the three-pronged test as required by the *Graham* factual inquires. There are significant differences between the cited references and the claimed invention as discussed above. Among other things, neither Teorey nor Farpinyo discloses converting logical aspects of CWM to design elements for relational database. In addition, Teorey does not refer to CWM at all. Furthermore, Farpinyo clearly does not consider the logical aspects of CWM as input for conversion.

Furthermore, the Examiner has not made an explicit analysis on the apparent reason to combine the known elements in the fashion in the claimed invention. Accordingly, there is no apparent reason to combine the teachings of Teorey and Farpinyo.

In the present invention, the cited references do not expressly or implicitly suggest any of the above elements. In addition, the Examiner failed to present a convincing line of reasoning as to why a combination of Teorey and Farpinyo is an obvious application of converting logical aspects of CWM to design elements for relational database, or an explicit analysis on the apparent reason to combine Teorey and Farpinyo in the manner as claimed.

Therefore, Applicant submits that claims 2-12, 14, 16-26, 28-38 are distinguishable over the cited prior art references.

VIII. CONCLUSION

Applicant respectfully requests that the Board enter a decision overturning the Examiner's rejection of all pending claims, and holding that the claims satisfy the requirements for patentability of 35 U.S.C. §101, 35 U.S.C. §102(b), and 35 U.S.C. §103(e).

Respectfully submitted,

UNISYS CORPORATION

Dated: August 13, 2007 By 

Phuong-Quan Hoang
Reg. No. 41,839
Tel.: 949-380-5643 (Pacific Coast)
Fax: 949-380-5254

Unisys Corporation
25725 Jeronimo Road, MS 400
Mission Viejo, CA 92691

IX. CLAIM APPENDIX

The claims of the present application which are involved in this appeal are as follows:

1. (original) A method comprising:

converting logical aspects of a common warehouse model (CWM) to corresponding design items for a relational database by processing in a hierarchical manner the logical aspects and creating the corresponding design items, the logical aspects comprising entity-relationship (ER) libraries, the ER libraries comprising ER models, the corresponding design items comprising design libraries, the design libraries comprising design models.

2. (original) The method of Claim 1 wherein converting comprises the operations of:

(a) scanning through the ER libraries;

(b) for a first of the ER libraries, creating a corresponding first design library;

(c) for each of the ER models in the first ER library, creating a corresponding design model in the corresponding first design library to hold corresponding information;

(d) processing each of the ER models to produce corresponding information for the corresponding design model;

(e) determining if there are any references between the ER models; and

(f) if there are any references between the ER models, specifying corresponding references in corresponding design models.

3. (original) The method of Claim 2 wherein, in operation (d), each of the ER models is processed independently.

4. (original) The method of Claim 1 wherein operation (d) comprises:

processing ER subject areas included in a first of the ER models;

processing ER domains included in the first ER model;
processing domain inheritance for each of the ER domains;
processing ER entities included in the first ER model;
processing entity subtype relationships in the first ER model; and
processing non-subtype relationships in the first ER model.

5. (original) The method of Claim 4 wherein processing ER subject areas comprises:

for each of the ER subject areas included in the first ER model, creating a corresponding design subject area in the corresponding first design model.

6. (original) The method of Claim 4 wherein processing domains comprises:

for each of the ER domains included in the first ER model, creating a corresponding design domain in the corresponding first design model;

determining parameters for each of the ER domains, including base type, default and constraint; and

setting corresponding parameters for each of the corresponding design domains.

7. (original) The method of Claim 4 wherein processing domain inheritance comprises:

determining, for a first of the ER domains, whether there is a first generalization in the CWM that links the first ER domain;

if there is the first generalization, determining parent ER domain and child ER domain for the first generalization, the parent and child ER domains corresponding to corresponding parent and child design domains; and

creating inheritance link from the corresponding child design domain to the corresponding parent design domain.

8. (original) The method of Claim 4 wherein processing ER entities comprises:

for a first ER entity included in the first ER model, creating a corresponding first design entity in the corresponding first design model;

determining first ER subject areas associated with the first ER entity, the first ER subject areas corresponding to first design subject areas;

adding the corresponding first design entity as a member of the corresponding first design subject areas; and

processing attributes associated with the first ER entity.

9. (original) The method of Claim 8 wherein processing attributes associated with the first ER entity comprises:

creating a first design attribute to correspond to the first ER attribute;

attaching the design attribute to the first design entity;

setting type reference of the first design attribute;

determining whether the first ER attribute is part of a first ER primary key associated with the first ER entity; and

if the first ER attribute is part of the first ER primary key, flagging the first design attribute as part of a first design primary key associated with the first design entity.

10. (original) The method of Claim 4 wherein processing entity subtype relationships comprises:

determining whether there is a first CWM generalization that links two of the ER entities in the first ER model;

if there is the first CWM generalization, determining parent and child ER entities for the first CWM generalization, the parent and child ER entities corresponding to corresponding parent and child design entities; and

creating inheritance link from the corresponding child design entity to the corresponding parent design entity.

11. (original) The method of Claim 4 wherein processing non-subtype relationships comprises:

obtaining references to parent and child ER entities in a first ER relationship, the parent and child ER entities corresponding to parent and child design entities in the first design model;

creating a corresponding design link between the corresponding parent and child design entities in the first design model;

setting cardinality and relationship type for the corresponding design link;

determining whether first ER relationship has at least one referential rule; and

if the first ER relationship has at least one referential rule, processing the at least one referential rule.

12. (original) The method of Claim 11 wherein processing the at least one referential rule comprises:

obtaining parameters including "insert", "update" and "delete" from the CWM;

setting corresponding parameters for the corresponding design link;

determining whether there is an ER attribute in the child ER entity that has migrated from the parent ER entity; and

if there is such an ER attribute corresponding to a design attribute, then:

creating a design foreign key under the child design entity; and

creating references to the corresponding design attribute.

13. (original) A method comprising:

converting logical aspects of a common warehouse model (CWM) to corresponding design items for a relational database by processing in a hierarchical manner the logical aspects and creating the corresponding design items, the logical aspects comprising entity-relationship (ER) libraries, the ER libraries comprising ER models, the corresponding design items comprising design libraries, the design libraries comprising design models; and

converting physical aspects of a common warehouse model (CWM) to corresponding database management system (DBMS) items in a relational database by

processing in a hierarchical manner the physical aspects and creating the corresponding DBMS items, the physical aspects comprising relational catalogs, the relational catalogs comprising relational schemas, the corresponding DBMS items comprising DBMS catalogs, the DBMS catalogs comprising DBMS schemas.

14. (original) The method of Claim 13 wherein converting logical aspects comprises the operations of:

(a) scanning through the ER libraries;

(b) for a first of the ER libraries, creating a corresponding first design library;

(c) for each of the ER models in the first ER library, creating a corresponding design model in the corresponding first design library to hold corresponding information;

(d) processing each of the ER models to produce corresponding information for the corresponding design model;

(e) determining if there are any references between the ER models; and

(f) if there are any references between the ER models, specifying corresponding references in corresponding design models;

and wherein converting physical aspects comprises:

(g) scanning through the relational catalogs;

(h) for a first of the relational catalogs, creating a corresponding first DBMS catalog in the relational database;

(i) for each of the relational schemas in the first relational catalog, creating a corresponding DBMS schema in the corresponding DBMS catalog to hold corresponding information; and

(j) processing each of the relational schemas to produce corresponding information for the corresponding DBMS schema.

15. (previously presented) An article of manufacture comprising:

a machine-accessible storage medium including data that, when accessed by a machine, cause the machine to perform the operation of:

converting logical aspects of a common warehouse model (CWM) to corresponding design items for a relational database by processing in a hierarchical manner the logical aspects and creating the corresponding design items, the logical aspects comprising entity-relationship (ER) libraries, the ER libraries comprising ER models, the corresponding design items comprising design libraries, the design libraries comprising design models.

16. (original) The article of manufacture of Claim 15 wherein the operation of converting comprises the operations of:

(a) scanning through the ER libraries;

(b) for a first of the ER libraries, creating a corresponding first design library;

(c) for each of the ER models in the first ER library, creating a corresponding design model in the corresponding first design library to hold corresponding information;

(d) processing each of the ER models to produce corresponding information for the corresponding design model;

(e) determining if there are any references between the ER models; and

(f) if there are any references between the ER models, specifying corresponding references in corresponding design models.

17. (original) The article of manufacture of Claim 16 wherein, in operation (d), each of the ER models is processed independently.

18. (original) The article of manufacture of Claim 15 wherein operation (d) comprises the operations of:

processing ER subject areas included in a first of the ER models;

processing ER domains included in the first ER model;

processing domain inheritance for each of the ER domains;

processing ER entities included in the first ER model;

processing entity subtype relationships in the first ER model; and

processing non-subtype relationships in the first ER model.

19. (original) The article of manufacture of Claim 18 wherein the operation of processing ER subject areas comprises:

for each of the ER subject areas included in the first ER model, creating a corresponding design subject area in the corresponding first design model.

20. (original) The article of manufacture of Claim 18 wherein the operation of processing domains comprises:

for each of the ER domains included in the first ER model, creating a corresponding design domain in the corresponding first design model;

determining parameters for each of the ER domains, including base type, default and constraint; and

setting corresponding parameters for each of the corresponding design domains.

21. (original) The article of manufacture of Claim 18 wherein the operation of processing domain inheritance comprises:

determining, for a first of the ER domains, whether there is a first generalization in the CWM that links the first ER domain;

if there is the first generalization, determining parent ER domain and child ER domain for the first generalization, the parent and child ER domains corresponding to corresponding parent and child design domains; and

creating inheritance link from the corresponding child design domain to the corresponding parent design domain.

22. (original) The article of manufacture of Claim 18 wherein the operation of processing ER entities comprises:

for a first ER entity included in the first ER model, creating a corresponding first design entity in the corresponding first design model;

determining first ER subject areas associated with the first ER entity, the first ER subject areas corresponding to first design subject areas;

adding the corresponding first design entity as a member of the corresponding first design subject areas; and

processing attributes associated with the first ER entity.

23. (original) The article of manufacture of Claim 22 wherein the operation of processing attributes associated with the first ER entity comprises:

creating a first design attribute to correspond to the first ER attribute;

attaching the design attribute to the first design entity;

setting type reference of the first design attribute;

determining whether the first ER attribute is part of a first ER primary key associated with the first ER entity; and

if the first ER attribute is part of the first ER primary key, flagging the first design attribute as part of a first design primary key associated with the first design entity.

24. (original) The article of manufacture of Claim 18 wherein the operation of processing entity subtype relationships comprises:

determining whether there is a first CWM generalization that links two of the ER entities in the first ER model;

if there is the first CWM generalization, determining parent and child ER entities for the first CWM generalization, the parent and child ER entities corresponding to corresponding parent and child design entities; and

creating inheritance link from the corresponding child design entity to the corresponding parent design entity.

25. (original) The article of manufacture of Claim 18 wherein the operation of processing non-subtype relationships comprises:

obtaining references to parent and child ER entities in a first ER relationship, the parent and child ER entities corresponding to parent and child design entities in the first design model;

creating a corresponding design link between the corresponding parent and child design entities in the first design model;

setting cardinality and relationship type for the corresponding design link;

determining whether first ER relationship has at least one referential rule; and

if the first ER relationship has at least one referential rule, processing the at least one referential rule.

26. (original) The article of manufacture of Claim 25 wherein the operation of processing the at least one referential rule comprises:

obtaining parameters including "insert", "update" and "delete" from the CWM;

setting corresponding parameters for the corresponding design link;

determining whether there is an ER attribute in the child ER entity that has migrated from the parent ER entity; and

if there is such an ER attribute corresponding to a design attribute, then:

creating a design foreign key under the child design entity; and

creating references to the corresponding design attribute.

27. (original) A system comprising:

a processor; and

a memory coupled to the processor, the memory containing program code that, when executed by the processor, causes the processor to perform the operation of:

converting logical aspects of a common warehouse model (CWM) to corresponding design items for a relational database by processing in a hierarchical manner the logical aspects and creating the corresponding design items, the logical aspects comprising entity-relationship (ER) libraries, the ER libraries comprising ER models, the

corresponding design items comprising design libraries, the design libraries comprising design models.

28. (original) The system of Claim 27 wherein the operation of converting comprises the operations of:

(a) scanning through the ER libraries;

(b) for a first of the ER libraries, creating a corresponding first design library;

(c) for each of the ER models in the first ER library, creating a corresponding design model in the corresponding first design library to hold corresponding information;

(d) processing each of the ER models to produce corresponding information for the corresponding design model;

(e) determining if there are any references between the ER models; and

(f) if there are any references between the ER models, specifying corresponding references in corresponding design models.

29. (original) The system of Claim 28 wherein, in operation (d), each of the ER models is processed independently.

30. (original) The system of Claim 27 wherein operation (d) comprises:

processing ER subject areas included in a first of the ER models;

processing ER domains included in the first ER model;

processing domain inheritance for each of the ER domains;

processing ER entities included in the first ER model;

processing entity subtype relationships in the first ER model; and

processing non-subtype relationships in the first ER model.

31. (original) The system of Claim 30 wherein the operation of processing ER subject areas comprises:

for each ER subject area included in the first ER model, creating a corresponding design subject area in the corresponding first design model.

32. (original) The system of Claim 30 wherein the operation of processing domains comprises:

for each of the ER domains included in the first ER model, creating a corresponding design domain in the corresponding first design model;

determining parameters for each of the ER domains, including base type, default and constraint; and

setting corresponding parameters for each of the corresponding design domains.

33. (original) The system of Claim 30 wherein the operation of processing domain inheritance comprises:

determining, for a first of the ER domains, whether there is a first generalization in the CWM that links the first ER domain;

if there is the first generalization, determining parent ER domain and child ER domain for the first generalization, the parent and child ER domains corresponding to corresponding parent and child design domains; and

creating inheritance link from the corresponding child design domain to the corresponding parent design domain.

34. (original) The system of Claim 30 wherein the operation of processing ER entities comprises:

for a first ER entity included in the first ER model, creating a corresponding first design entity in the corresponding first design model;

determining first ER subject areas associated with the first ER entity, the first ER subject areas corresponding to first design subject areas;

adding the corresponding first design entity as a member of the corresponding first design subject areas; and

processing attributes associated with the first ER entity.

35. (original) The system of Claim 34 wherein the operation of processing attributes associated with the first ER entity comprises:

creating a first design attribute to correspond to the first ER attribute;

attaching the design attribute to the first design entity;

setting type reference of the first design attribute;

determining whether the first ER attribute is part of a first ER primary key associated with the first ER entity; and

if the first ER attribute is part of the first ER primary key, flagging the first design attribute as part of a first design primary key associated with the first design entity.

36. (original) The system of Claim 30 wherein the operation of processing entity subtype relationships comprises:

determining whether there is a first CWM generalization that links two of the ER entities in the first ER model;

if there is the first CWM generalization, determining parent and child ER entities for the first CWM generalization, the parent and child ER entities corresponding to corresponding parent and child design entities; and

creating inheritance link from the corresponding child design entity to the corresponding parent design entity.

37. (original) The system of Claim 30 wherein the operation of processing non-subtype relationships comprises:

obtaining references to parent and child ER entities in a first ER relationship having ends, the parent and child ER entities corresponding to parent and child design entities in the relational database;

creating corresponding link between the corresponding parent and child design entities in the relational database;

setting cardinality and relationship type for the corresponding link;

determining whether the ends of the first ER relationship have at least one referential rule;

if the ends of the first ER relationship have at least one referential rule, processing the at least one referential rule.

38. (original) The system of Claim 11 wherein the operation of processing the at least one referential rule comprises:

obtaining parameters including "insert", "update" and "delete" from the CWM;

setting corresponding parameters for the corresponding design link;

determining whether there is an ER attribute in the child ER entity that has migrated from the parent ER entity; and

if there is such an ER attribute corresponding to a design attribute, then:

creating a design foreign key under the child design entity; and

creating references to the corresponding design attribute.

XI. EVIDENCE APPENDIX

None

XII. RELATED PROCEEDINGS APPENDIX

None